INTRODUCERE

Interfata 1, pe scurt IF1, inglobeaza de fapt pe o singura placuta de circuit imprimat trei interfete: cu discul flexibil (sau floppy), cu o linie seriala standard CCITT V24 si o interfata mai putin obisnuita care permite cuplarea mai multor HC-uri printr-o singura pereche de fire torsadate, conexiune denumita retea.

Interfata de floppy permite cuplarea a pina la doua minidrive-uri (5.25 toli) cu 35 sau 40 de piste realizind o capacitate formata de 140 respectiv 160 kilookteti pe fiecare floppy disc, spatiu care poate fi utilizat pentru a memora pina la 64 de fisiere distincte.

Rata medie de transfer a discului este de 25 pina la 30 de ori mai mare decit a interfetei standard de caseta magnetica. Daca mai adaugam la aceasta si accesul aleator la informatii (timpul maxim de acces la un sector de disc este de circa 1.7 secunde), este imposibil sa nu remarcam avantajele majore fata de interfata de caseta.

Interfata seriala rezolva, in principal, problema cuplarii unei imprimante la HC85, dar poate fi folosita si pentru a transfera date cu orice alt tip de calculator.

Interfata de retea ofera o solutie pentru una din aplicatiile posibile ale calculatorului HC85: invatamintul. Cuplate intr-o retea de pina la 64 de sisteme, reteaua HC-urilor dintr-o sala de laborator informatic poate usura atit sarcina profesorului, cit si sarcina elevilor.

Viteza de transfer a informatiei prin retea este de 80 kilookteti pe secunda. Transferurile de date se fac in blocuri cu lungime variabila (maxim 255 octeti), insotite de blocuri de control care specifica adrese sursa/destinatie, numar bloc, etc. Protocolul este suficient de cuprinzator pentru a permite schimbul simultan de mesaje intre oricite noduri ale retelei folosind numai doua fire torsadate pentru a lega nodurile intre ele.

Din punct de vedere programe, IF1 se integreaza in sistemul BASIC al calculatorului HC85, oferind fie noi instructiuni, fie extensii ale instructiunilor existente. Extinderea limbajului BASIC se face fara nici o modificare a placii de baza, interfata 1 interceptind prin hardware rutina de eroare din placa de baza.

Cele trei interfete ofera pe linga facilitati de incarcare salvare programe si date, comenzi pentru manipularea de fisiere, ceea ce ofera o noua dimensiune in stocarea si regasirea datelor folosind programe scrise in BASIC.

PRIMELE OPERATII CU MINIDISCUL

Auto-run

Dupa ce ati terminat de instalat Interfata 1 si Minidrive-ul sinteti curios sa aflati ce programe va asteapta pe discul de demonstratie. Pentru asta, inserati floppy discul in Minidrive (sau daca aveti doua Minidrive-uri, in Minidrive-ul 1), si introduceti:

```
NEW
```

urmat de:

```
RUN (si CR)
```

Aceste comenzi vor declansa incarcarea automata si rularea primului program de pe floppy. Dupa ce ati terminat de privit acest program, cititi mai departe.

Catalogul

Pentru a afla ce alte programe se gasesc pe floppy-ul de demonstratie, introduceti intructiunea CATalog:

```
CAT 1
    !
    +---  1 identifica numarul Minidrive-ului pe care
          il folositi
```

In aproximativ 3 secunde pe ecranul televizorului se va afisa:

- un catalog al tuturor numelor fisierelor memorate pe floppy
- spatiul ramas disponibil pe floppy (in kiloocteti)

Incarcarea programelor

Urmatorul lucru de facut este incarcarea programului pe care vreti sa-l executati in continuare. Pentru asta alegeti mai intii un program, apoi introduceti:

```
LOAD *"d";1;"nume"
     ! !      !
     ! !      +---    aici introduceti numele programului
     ! !              pe care l-ati ales
     ! !
     ! +---------     "d";1 identifica ce Minidrive
     !                folositi
     !
     +-----------     steluta comunica calculatorului ca
                      folositi un Minidrive, si nu
                      interfata obisnuita de caseta
```

Dupa o scurta pauza, ecranul va afisa mesajul OK (dar fara numele programului). Puteti acum lansa programul in executie (cu RUN).

UTILIZARE MINIDISC PENTRU PROGRAME

Salvare, verificare, incarcare si comasare programe

In manualul de utilizare HC85 ati gasit instructiunea SAVE. Aceasta salveaza programe pe caseta. Salvarea programelor pe floppy este la fel de simpla. Pentru exemplificare va fi folosit programul de mai jos, denumit Patrate. El tipareste numerele de la 1 la 10 impreuna cu patratele lor.

```
10 REM Patrate
20 FOR n=1 TO 10
30 PRINT n,n*n
40 NEXT n
```

Pentru a salva acest program pe caseta, ati fi introdus:

```
SAVE "Patrate"
```

Pentru a-l salva pe floppy-ul din Minidrive-ul 1, introduceti:

```
SAVE *"d";1;"Patrate"
```

Dupa citeva secunde in care marginea ecranului va clipi, programul va fi salvat.

(Numele programelor memorate pe floppy pot avea o lungime maxima de 11 caractere).

Asa cum probabil v-ati imaginat deja, puteti verifica corecta inregistrare a programului pe floppy introducind:

```
VERIFY *"d";1;"Patrate"
```

Ecranul va afisa mesajul OK.

Puteti incarca acum programul Patrate introducind:

```
NEW
```

urmat de:

```
LOAD *"d";1;"Patrate"
```

In continuare, pentru a face ca programul sa se lanseze automat, incercati sa introduceti:

```
SAVE *"d";1;"Patrate2" LINE 10
```

apoi:

```
NEW
```

si apoi:

```
LOAD *"d";1;"Patrate2"
```

Minidrive-ul poate fi folosit si pentru a comasa programele. Introduceti:

```
NEW
```

urmat de:

```
100 REM alte Patrate
110 FOR n=11 TO 20
120 PRINT n,n*n
130 NEXT n
```

si acum introduceti:

```
MERGE *"d";1;"Patrate"
```

si programul Patrate va fi adaugat la listing.

Pe scurt, asa cum v-ati dat deja seama, sintaxa folosita pentru obisnuita interfata de caseta (explicata in sectiunea 3.20 din manualul de programare BASIC) se aplica si la Minidisc.

Stergerea programelor

Sa presupunem ca ati terminat de lucrat cu programul Patrate. Pentru a-l sterge, introduceti:

```
ERASE "d";1;"Patrate"
```

(Ca inainte, "d";1 indica ce Minidrive folositi).

In timpul executiei intructiunii ERASE, marginea ecranului va clipi.

Formarea discurilor

Inainte de prima utilizare a unui floppy, inserati-l intr-un Minidrive (de exemplu Minidrive-ul 1) si introduceti:

```
FORMAT "d";1
```

"d";1 identifica Minidrive-ul pe care il folositi (in acest caz Minidrive-ul 1).

Formarea unui disc dureaza aproximativ treizeci de secunde. In timpul acesta, marginea ecranului se va schimba la inceput, si va reveni putin inainte de afisarea mesajului OK. Procesul de formare consta din initializarea fiecarei piste de pe floppy, prin scrierea cimpurilor de identificare si date corespunzatoare fiecarui sector. Dupa formarea unei piste, fiecare sector in parte este citit, verificind suma de control. Mesajul OK apare numai daca toate sectoarele au putut fi citite corect (nu se accepta discuri cu sectoare eronate).

Formarea unui floppy nu trebuie repetata niciodata, si pentru ca prin formarea unui disc se pierde orice a fost inregistrat pe el.

Apasati acum:

```
          CAT 1
              |
              +---  1 identifica numarul minidrive-ului pe care
                    il folositi
```

Dupa citeva secunde, in care timp marginea ecranului va clipi, va apare mesajul de eroare:

```
          File not found
```

care semnifica faptul ca floppy-ul nu contine nici un program.

Capacitatea unui disc este de 158 kiloocteti, plus 2 kiloocteti pentru zona de catalog, gestionata de sistem.

Instalarea facilitatii de auto-run

Putin mai inainte ati folosit facilitatea de auto-run pentru discul de demonstratie. Daca aveti un program pe care il folositi adesea, va puteti stabili propria facilitate de auto-run, astfel incit sa nu mai introduceti instructiunile LOAD si RUN. Acestea sint regulile de urmat:

- programul trebuie sa aiba numele run;
- floppy-ul trebuie folosit in Minidrive-ul 1;
- facilitatea trebuie folosita fie imediat dupa punerea sub tensiune, sau imediat dupa comanda NEW.

Astfel, introduceti programul respectiv, urmat de comanda:

```
          SAVE *"d";1;"run" LINE numar
                      |           |
                      |           +--- introduceti aici numarul
                      |                liniei de start
                      |
                      +--------   numele run trebuie introdus
                                  litera cu litera.  Nu apasati
                                  tasta RUN
```

Acum introduceti:

```
          NEW
```

urmat de:

```
          RUN
           |
           +---  Tasta RUN, si nu numele programului.
```

DATE, CANALE SI CAI

Precum stiti, un program este un set de operatii care se executa atunci cind apasati RUN. Datele, pe de alta parte, sint orice colectie de litere, numere sau simboluri cu care poate lucra un program. Exemple sint numerele de la 1 la 10 si patratele lor.

Datele pot fi trimise, sau receptionate, catre/de la diferite parti ale unui sistem de calcul. Aceste parti sint denumite 'canale'. Canalele catre care se pot trimite date sint:

- ecranul televizorului

- un fisier pe floppy

- un alt calculator HC85, daca amindoua calculatoarele sint cuplate printr-o retea.

- interfata RS232 si de acolo, de exemplu, la un modem sau o imprimanta.

Canalele de la care se pot primi date sint:

- claviatura

- un fisier pe floppy

- un alt calculator HC85, daca amindoua calculatoarele sint cuplate printr-o retea.

- interfata RS232, adica un modem sau un terminal.

Nodurile de comunicatie dintre programul BASIC si canale sint denumite cai. In sistemul HC85, numarul acestor cai este fixat la 16. Ele sint numerotate de la 0 la 15, iar numerele de cale sint intotdeauna precedate de semnul #.

Patru dintre aceste cai sint deja cuplate la urmatoarele canale:

calea #0	trimite date catre partea de jos a ecranului TV si
calea #1	primeste date de la claviatura;
calea #2	trimite date catre partea de sus a ecranului TV, dar nu poate primi date;
calea #3	trimite date catre imprimanta, dar nu poate primi date.

Orice intructiune care executa un transfer de intrare/iesire foloseste una din aceste cai in mod implicit. De exemplu, instructiunea PRINT foloseste calea #2, iar instructiunea LPRINT foloseste calea #3. Astfel, daca introduceti:

PRINT "Acesta este un calculator HC85"

folositi de fapt o prescurtare a instructiunii:

PRINT #2;"Acesta este un calculator HC85"

Verificati prin introducerea celor doua forme.

Puteti, totusi, sa faceti fiecare instructiune sa foloseasca o alta cale prin introducerea semnului # urmat de un numar de cale. Incercati sa introduceti:

LPRINT #2;"Acesta este un calculator HC85"

in loc sa fie trimis la imprimanta, acest mesaj apare pe ecranul TV.

Dar in loc sa folositi caile prestabilite, puteti crea unele proprii. Caile #4 pina la #15 sint rezervate pentru acest scop; si exista diferite specificatoare de canale care indica perifericul dorit. Citeva exemple sint:

"K" pentru claviatura

"S" pentru ecran

"P" pentru imprimanta

(altele vor fi introduse mai tirziu).

Remarcati faptul ca K, S si P sint toate canale prestabilite. Ele solicita utilizarea virgulelor (,) drept separatori in instructiunile OPEN #. Dar cu alte canale puteti folosi fie virgule fie punct-virgula (;).

Pentru a creea o cale proprie folositi instructiunea OPEN #. De exemplu introduceti:

10 OPEN #4,"S"

Astfel deschideti calea #4 si il cuplati la canalul "S". Acum introduceti:

20 PRINT #4;"Acesta este un calculator HC85"

si din nou linia va apare pe ecran.

(Nu se recomanda deschiderea cailor 0, 1 sau 2, pentru ca rezultatele acestor operatii pot fi imprevizibile).

FISIERE DE DATE PE DISC

Deschiderea unui fisier de date

Memorarea informatiilor pe floppy se face in fisiere. Fiecare fisier primeste la creeare un nume, pentru a putea fi regasit mai tirziu. Instructiunea care deschide si denumeste un fisier de date are intotdeauna aceeasi forma. De exemplu testati instructiunea:

```
OPEN #4;"d";1;"Numere"
      !    !     !
      !    !     +---    "Numere" este numele fisieru-
      !    !             lui. Acesta poate fi orice
      !    !             sir de caractere de lungime
      !    !             maxima 11
      !    !
      !    +---------    "d";1 identifica Minidiscul pe
      !                  care il folositi
      !
      +--------------    numarul de cale poate fi orice
                         numar intre 0 si 15
```

Aceasta instructiune face doua lucruri distincte:

- stabileste un canal cuplat cu fisierul: "d";1;"Numere"

- ataseaza acest nou canal la calea #4.

Operatia va dura citeva secunde, in care timp calculatorul va cauta pe floppy un fisier cu numele "Numere". Pentru ca nu exista fisierul "Numere", deschide canalul pentru scriere. (Daca ar fi gasit un fisier cu numele "Numere", l-ar fi deschis pentru citire).

Introducerea datelor

Odata ce ati deschis un fisier, puteti introduce date. Sa presupunem ca vreti sa memorati numerele de la 1 la 10 impreuna cu patratele lor. Introduceti si rulati programul urmator:

```
10 FOR n=1 TO 10
20 PRINT #4;n'n*n
30 NEXT n
```

S-ar putea sa credeti ca toate numerele au fost deja memorate pe floppy. Dar de fapt calculatorul nu transfera in mod automat datele pe floppy decit dupa ce s-a acumulat o anumita cantitate de informatii, pe care o transfera dintr-odata. Acest procedeu se numeste 'blocarea' datelor. Un bloc de date pe floppy are lungimea de 256 de octeti (sau caractere).

Pentru a memora pe floppy datele introduse trebuie sa inchideti fisierul. Pina nu faceti acest lucru, nu veti putea sa cititi din fisier.

Inchiderea unui fisier

Inchiderea unui fisier asigura memorarea definitiva a datelor pe floppy. Inchide de asemenea canalul (in cazul nostru "d";1;"Numere") si detaseaza calea (in cazul nostru #4) de la orice canal. Pentru a inchide un fisier trebuie doar sa inchideti calea asociata:

```
CLOSE #4
```

Marginea ecranului va clipi pentru a arata ca se inregistreaza ceva pe floppy.

(Remarcati faptul ca, la fel ca la instructiunea OPEN, instructiunea CLOSE este urmata in mod automat de #).

Caile #0, #1, #2, #3 ramin intotdeauna atasate unui canal, chiar daca se executa o instructiune CLOSE specifica. Daca incercati sa inchideti una din aceste cai, caile #0 si #1 se vor atasa automat la canalul K; calea #2 la canalul S; iar calea #3 la canalul P.

Citirea datelor dintr-un fisier

Pentru a citi datele din fisierul "Numere" rulati urmatorul program:

```
10 OPEN #4;"d";1;"Numere"
20 FOR b=1 TO 10
30 INPUT #4;m;n
40 PRINT "Patratul lui ";m;" este ";n
50 NEXT b                 ¦ `   ¦    ¦
60 CLOSE #4               +-----+----+--- lasati spatii
RUN
```

Pentru ca fisierul "Numere" exista deja pe floppy, canalul "d";1;"Numere" este deschis pentru intrare, si orice incercare de a scrie date ar fi generat o eroare.

Se poate de asemenea folosi functia INKEY$ pentru a citi date dintr-un fisier (intoarce intotdeauna urmatorul caracter din fisier). Incercati programul urmator:

```
10 OPEN #11;"d";1;"listing"
20 LIST #11
30 CLOSE #11
40 OPEN #12;"d";1;"listing"
50 PRINT INKEY$#12;
60 GO TO 50
```

Acest program se va termina cu un mesaj de sfirsit de fisier, adica End of file.

Observatii asupra lui PRINT si INPUT

Pentru ca instructiunile PRINT si INPUT au fost concepute in principal pentru utilizarea cu ecranul si claviatura, trebuie sa fiti atenti la folosirea lor cu fisiere.

'separatori'

Instructiunea PRINT are trei forme de separatori:

- semnul ; (punct-virgula) nu tipareste nimic,
- semnul , (virgula) va aduce la inceputul urmatoarei jumatati de linie,
- semnul ' (apostrof) sare la linie noua (codul CR).

Instructiunea INPUT asteapta intotdeauna sa introduceti CR dupa un numar sau un sir. Astfel, de fiecare data cind tipariti intr-un fisier din care vreti sa cititi mai tirziu cu INPUT, trebuie fie sa:

- tipariti fiecare element separat, adica

```
10 PRINT #4;2
20 PRINT #4;3
```

sau

- separati elementele cu apostrof, adica

```
10 PRINT #4;2'3
```

De asemenea, in instructiunile INPUT, trebuie sa folositi cu atentie separatorii. Asa cum stiti, INPUT poate tipari in partea de jos a ecranului orice se poate pune intr-o instructiune PRINT. Dar daca cititi cu INPUT dintr-un fisier, fisierul se deschide numai pentru citire. Asa incit, daca includeti orice s-ar fi tiparit la utilizarea ecranului, veti obtine mesajul de eroare **Writing to a 'read' file** (Scriere intr-un fisier de citire). Aceasta inseamna ca elementele dintr-o instructiune INPUT trebuie separate numai prin punct-virgula, adica

```
10 INPUT #4;a;b
```

Atentie de asemenea la citirea cu INPUT a sirurilor de caractere care contin " (ghilimele), pentru ca INPUT va interpreta ghilimelele drept sfirsit de sir. Metoda de a evita acest lucru este de a inlocui, de exemplu:

```
10 INPUT #4;a$
```

cu

```
10 INPUT #4; LINE a$
```

Schimbarea cailor

Instructiunile PRINT pot contine informatii pentru mai multe cai la un moment dat. Programul urmator va tipari "unu" pe ecran; "doi" intr-un fisier pe floppy denumit "cifre"; "trei" catre statia 1 pe retea (vezi capitolul urmator); si "patru" in urmatoarea linie din ecran.

```
10 OPEN #4;"d";1;"cifre"
20 OPEN #5;"n";1
30 PRINT "unu";#4;"doi";#5;"trei"'#2;"patru"
40 CLOSE #4
50 CLOSE #5
```

'schimbarea culorilor'

Dupa ce ati folosit un canal diferit de ecran, se poate ca instructiunile PAPER si INK sa nu aiba nici un efect. Pentru a evita acest lucru, introduceti:

```
PRINT;
```

inainte de a schimba PAPER sau INK.

Afisarea catalogului de fisiere

Pe masura ce se inregistreaza fisiere pe floppy, acestea sint introduse in mod automat in catalog. Astfel, pentru a afla ce fisiere sint inregistrate pe floppy, este sufiecient sa inserati floppy-ul intr-un Minidrive si sa introduceti instructinuea CATalog. De exemplu, introduceti:

```
CAT 1
```

Ecranul televizorului va afisa:

- numele fisierelor

- spatiul disponibil ramas pe floppy (in kilooctet)

Puteti sa transferati iesirea unui CAT catre o cale introducind:

```
CAT #numar;numar
      |     |
      |     +-----  numar Minidrive
      |
      +----------   numar cale
```

Aceasta va permite sa trimiteti catalogul catre o imprimanta, sau catre un fisier, astfel incit sa poata fi folosit de un program.

Protejarea unui fisier

Daca doriti ca un nume sa nu apara in catalog, il puteti proteja dindu-i un nume care are in pozitia 10 codul caracterului dorit plus 128. Introduceti acest program:

```
10 OPEN #4,"d";1;"Rezultate"+ CHR$ (128+ CODE " "
20 FOR n=1 TO 15
30 PRINT #4;n'n*n
40 NEXT n
50 CLOSE #4
```

Acum introduceti:

```
CAT 1
```

Numele fisierului nu va apare. Astfel ca, de fiecare data cind generati un fisier protejat, amintiti-va sa-i notati numele undeva, pentru cazul in care ii uitati numele!

Extinderea unui fisier

Sa presupunem ca vreti sa extindeti fisierul "Numere" pentru a include patratele numerelor de la 1 la 20 in loc de numai 1 la 10. Un fisier nu poate fi redeschis pentru scriere, astfel ca trebuie sa:

- creati o noua versiune cu alt nume;

- transferati vechiul fisier in noua versiune;

- adaugati noile date

- inchideti vechiul fisier.

Iata cum se poate face aceasta.

Mai intii rulati acest program:

```
 10 OPEN #4;"d";1;"Numere": REM pentru citire
 20 OPEN #5;"d";1;"Numere 1": REM pentru scriere
 30 FOR f=1 TO 10
 40 INPUT #4;m;n
 50 PRINT #5;m'n
 60 NEXT f
 70 FOR n=11 TO 20
 80 PRINT #5;n'n*n
 90 NEXT n
100 CLOSE #4; CLOSE #5
```

Pentru a verifica existenta a doua fisiere, "Numere" si "Numere 1", introduceti:

```
CAT 1
```

Apoi, ca sa stergeti vechiul fisier, introduceti:

```
ERASE "d";1;"Numere"
```

Pentru a verifica stergerea, introduceti:

```
CAT 1
```

Numele fisierului "Numere" a disparut din catalog, iar noul fisier, "Numere 1" contine acum numerele de la 1 la 20.

RETEAUA LOCALA

Configurarea unei retele

Reteaua locala permite utilizatorului si prietenilor lui sa schimbe intre ei programe si date. Aceasta inseamna ca numai unul dintre voi trebuie sa introduca un program. O retea este foarte utila si daca numai unul dintre voi are un Minidisc.

Folosind cablul furnizat odata cu interfata, puteti lega de la doua pina la saizecisipatru de calculatoare HC85.

Configuratia retelei nu trebuie sa fie in nici un caz o bucla inchisa: calculatoarele de la capetele retelei nu trebuie sa fie conectate intre ele. Fiecare capat de retea trebuie sa aiba un conector neocupat.

```
===================================================================
!                                                                 !
!  NU PORNITI SI NU OPRITI NICIODATA UN HC85 CARE ESTE CUPLAT     !
!  LA RETEA IN TIMP CE ARE LOC UN TRANSFER DE DATE PE RETEA.      !
!  Totusi puteti avea un HC85 oprit pe retea; puteti de aseme-    !
!  nea sa porniti sau sa opriti HC85-uri care sint pe retea,      !
!  cu conditia ca sa nu se faca transferuri pe retea in acel      !
!  moment.                                                        !
!                                                                 !
===================================================================
```

Dupa ce ati stabilit o retea, fiecare calculator (sau statie) trebuie sa primeasca un numar de identificare diferit. Mai intii stabiliti impreuna cu prietenii dumneavoastra, care va fi numarul fiecarei statii, dupa care fiecare dintre dumneavoastra trebuie sa introduca:

```
FORMAT "n";numar
            !
            +-------    introduceti numarul de statie
                        pe care l-ati ales.
```

Daca reteaua este formata numai din doua calculatoare, amindoua pot folosi acelasi numar de statie. Si pentru ca, amindoua calculatoarele devin statia 1 in mod automat la punerea sub tensiune, utilizarea instructiunii FORMAT nu mai este necesara.

Programele si reteaua

Sa presupunem ca ati cuplat doua calculatoare intr-o retea, cu numerele de statie 1 respectiv 2.

Sa presupunem ca vreti sa trimiteti catre statia 2 urmatorul program:

```
10 REM patrate
20 FOR n=1 TO 10
30 PRINT n,SQR n
40 NEXT n
```

Introduceti programul urmat de:

```
SAVE *"n";2
```

(Remarcati ca reteaua nu foloseste nume pentru programe.)

Intre timp la statia 2 trebuie introdus:

```
FORMAT "n";2
```

urmat de:

```
LOAD *"n";1
```

Statia 2 va avea acum o copie a programului. Remarcati cum marginea ecranului ramine neagra in timp ce calculatorul asteapta sa salveze sau sa incarce programul prin retea. Statia 1 nu va trimite pina cind statia 2 nu este gata, iar statia 2 va astepta pina cind se emite ceva. Incercati sa introduceti linia cu SAVE inainte ca la statia 2 sa introduceti LOAD si vice versa.

Pentru a verifica transmisia corecta a programului, la statia 2 trebuie introdus:

```
VERIFY *"n";1
```

in timp ce la statia 1 se repeta transmisia programului introducind:

```
SAVE *"n";2
```

SAVE este de fapt singura instructiune care transmite programe in retea. Instructiunile LOAD, VERIFY si MERGE sint toate metode de a receptiona programele.

Jocul de retea din Apendixul 1 este un bun exemplu pentru utilizarea programelor in retea.

Fisiere de date in retea

Sa presupunem ca doriti sa transmiteti acum date catre statia 2. Instructiunea OPEN #4;"n";2 deschide un canal catre statia 2 pe retea si ataseaza calea #4 la el, astfel incit daca scrieti prin calea #4, mesajul va fi pus pe retea impreuna cu o notita care indica sursa mesajului.

Daca ati fi introdus INPUT #4;m$ calculatorul dumneavoastra ar fi asteptat informatii adresate statiei 1 de la statia 2.

Acum introduceti acest program:

```
10 OPEN #4;"n";2: REM pentru iesire
20 INPUT a$: PRINT #4;a$
30 CLOSE #4
40 OPEN #4;"n";2: REM pentru intrare
50 INPUT #4;b$: PRINT b$
60 CLOSE #4
70 GO TO 10
```

Apoi introduceti:

```
SAVE *"n";2
```

Acum introduceti la statia 2:

```
FORMAT "n";2
LOAD *"n";1
```

Introduceti la statia 1 RUN, iar la statia 2 editati liniile 10 si 40 pentru a se referi la statia 1 si nu la statia 2. Apoi se introduce la statia 2:

```
GO TO 40
```

Sinteti acum gata sa incepeti o conversatie. Dar inainte de a face asta ar trebui sa stiti citeva lucruri.

- Tot ce tipariti prin calea #4 este blocat: adica nu este imediat pus pe retea, ci se asteapta pina cind se acumuleaza o anumita cantitate de date. Asa ca este necesara inchiderea canalului prin CLOSE, imediat ce ati terminat de tiparit. Astfel se transmite zona tampon chiar daca nu este plina. (Zona tampon are lungimea 255 de octeti sau caractere.)

- Tot ce tipariti este marcat ca sosind in mod specific de la statia la care lucrati, astfel incit daca statia 2 este in asteptare pentru un mesaj de la alta statie, mesajul dumneavoastra va fi ignorat. Daca mesajul emis este ignorat, ecranul nu va afisa mesajul OK, si marginea ecranului va ramine neagra pina cind mesajul este emis si se primeste confirmare pozitiva de la statia 2.

- In timp ce instructiunea INPUT poate fi folosita pentru a astepta ca ceva sa fie transmis, functia INKEY$ poate fi folosita pentru a citi reteaua. Se va intoarce cu primul octet din orice a fost transmis, sau din orice asteapta sa fie transmis. Altfel se intoarce cu sirul vid. Aceasta se numeste interogare (polling).

Programul de mai jos va tipari orice este transmis catre statia 1:

```
10 OPEN #8;"n";1
20 PRINT INKEY$#8;
30 GO TO 20
```

(Pentru mai multe detalii despre INKEY$ vezi sectiunea 3.15 din manualul tehnic HC 85.)

Emisie generala

Exista un numar de statie special, al carui specificator este "n";0. Atunci cind se asteapta date de la statia 0, veti receptiona orice mesaj care este emis catre statia 0. Iar atunci cind scrieti, mesajul emis catre statia 0 va fi receptionat de oricine citeste date de la un canal cu specificatorul "n";0.

Aceasta ar fi foarte util, de exemplu, intr-o scoala daca fiecare elev ar avea un calculator, dar numai profesorul ar avea un minidrive.

Sa presupunem ca profesorul doreste sa emita un program. Mai intii toti elevii ar trebui sa introduca:

```
LOAD *"n";0
```

Aceasta comanda va face ca toate calculatoarele elevilor sa intre in asteptare pentru receptia programului. Profesorul ar trebui sa salveze programul in retea introducind:

```
SAVE *"n";0
```

Emisiile generale, spre deosebire de mesajele private, incep imediat fara sa astepte ca alte calculatoare sa fie gata sa le receptioneze.

De asemenea, la emisia generala, calculatorul nu va poate informa daca mesajul emis a fost receptionat de catre cineva.

Functia INKEY$ nu poate fi folosita pentru a interoga un canal de receptie generala. La fel ca INPUT, ea va astepta pur si simplu ca ceva sa fie emis.

UTILIZAREA INTERFETEI SERIALE

Conectarea perifericelor la interfata seriala

Precum stiti, setul de caractere al lui HC85 contine atit simboluri simple (litere, cifre, etc.) cit si cuvinte cheie (instructiuni, nume de functii, etc.). Toate aceste caractere pot fi emise si receptionate prin interfata seriala catre/de la orice dispozitiv compatibil; de exemplu o imprimanta, un modem sau o alta interfata seriala conectata la un tip diferit de calculator.

Pentru a conecta oricare din aceste periferice la interfata seriala, trebuie sa folositi un cablu cu un conector cu 9 pini la capatul dinspre interfata 1 si un conector corespunzator dispozitivului la care va cuplati la celalalt capat. (Pentru detalii de interconectare vezi Apendixul 4.)

Apoi, inainte de a folosi interfata seriala, va trebui sa stabiliti modul de lucru al perifericului:

- modul 'auto line feed' trebuie dezactivat. (HC85 va emite secventa 'retur car' (CR) si 'avans rind' (LF) pe un canal "t", dar numai 'retur car' (CR) pe un canal "b". Aceste canale "t" si "b" sint explicate mai jos.)

- paritatea trebuie dezactivata.

- numarul de biti trebuie stabilit la 8 (opt).

- numarul de biti de stop trebuie stabilit la 1 (unu).

- viteza de emisie/receptie (adica numarul de biti pe secunda). HC85 poate comunica la oricare din vitezele standard, adica:

  50, 110, 300, 600, 1200, 2400, 4800, 9600, 19200.

  Este bine sa folositi cea mai mare viteza pe care o permite perifericul la care va cuplati. (Veti vedea mai jos cum puteti face ca HC85 sa foloseasca aceasi viteza.)

In astfel de momente, un manual de instalare pentru perifericul la care va cuplati este foarte util.

Canalele t si b

Interfata seriala are doua tipuri de canale: canalul t si canalul b.

Canalul t

Canalul t (de la text) este folosit de obicei pentru a trimite listinguri. Canalul t are urmatorul efect asupra setului de caractere:

cod caracter	
0-31	(caracterele de control) nu sint emise, cu exceptia lui 13 (retur car) care este trimis ca 13 urmat de 10 (retur car si avans rind).

32-126 (caracterele tiparibile) sint trimise ca atare.

127-164 (caracterele grafice) nu sint emise. Ele sint inlocuite de caracterul ? (codul 63).

Pentru INPUT si INKEY$ canalul t transfera numai caractere pe 7 biti, asa incit forteaza la 0 bitul 7.

Pentru a folosi canalul t, trebuie mai intii sa stabiliti viteza de lucru pe sirme. Asa ca introduceti:

```
10 FORMAT "t";viteza
               !
               +-----   introduceti aici viteza de
                        lucru pe care ati stabilit-o
                        si la periferic.
```

Acum, pentru a deschide o cale catre canalul t, introduceti:

```
20 OPEN #3,"t"
30 LLIST
```

Marginea ecranului va clipi si listingul va fi trimis catre periferic. (Remarcati ca LLIST este este o prescurtare de la LIST #3.) Introduceti acum:

```
LPRINT "Acesta este un mesaj."
```

Si acest mesaj va fi trimis catre dispozitiv.

Daca HC85 este cuplat cu un terminal sau un calculator care poate trimite caractere, atunci puteti citi date de la terminal sau calculator. Introduceti:

```
10 FORMAT "t";viteza
20 OPEN #4,"t"
30 PRINT INKEY$#4;
40 GO TO 30
```

Acum, orice caracter primit de la terminal sau calculator va fi afisat pe ecran.

Canalul b

Canalul b (de la binar) trimite toti cei 8 biti ai codurilor folosite de HC85, si va permite sa trimiteti coduri de control catre imprimante etc.

Si la INPUT si INKEY$ canalul b intoarce caractere pe 8 biti.

SAVE si LOAD functioneaza numai cu canalul b.

Daca ati conectat doua HC-uri prin interfata seriala sau doriti sa va memorati programele pe un alt tip de calculator care are deasemenea o interfata seriala, veti dori sa salvati si sa incarcati programe prin interfata seriala. Pentru aceasta introduceti:

```
FORMAT "b",viteza
           !
           +---- introduceti aici viteza pe care ati
                 stabilit-o la periferic
```

Acum puteti incerca, de exemplu:

```
10 REM cifre
20 FOR n=1 TO 10
30 PRINT n,n*RND
40 NEXT n
```

urmat de:

```
SAVE *"b"
```

La celalalt capat al legaturii cineva trebuie sa introduca:

```
LOAD *"b"
```

Extensiile uzuale sint deasemenea posibile:

```
SAVE *"b";SCREEN$
```

si:

```
SAVE *"b";LINE numar
```

Cum se trimit coduri de control

Multe imprimante primesc secvente de control pentru operatii de genul tiparire cu latime dubla. Pentru a trimite caracterele de control trebuie sa folositi canalul b. Atentie insa, prin canalul b returul de car (CR) nu este urmat automat de avans-rind (LF). De aceea este preferabil sa aveti doua canale deschise, unul b si altul t: veti folosi canalul b pentru a trimite secventele de control, si canalul t pentru a trimite textele.

De exemplu sa presupunem ca codul de control pentru tiparire cu latime dubla este 14. Introduceti:

```
10 OPEN #4;"b"
20 PRINT #4;"Latime normala ";
30 PRINT #4;CHR$ 14;"Latime dubla"
40 CLOSE #4
```

(Daca exemplul nu functioneaza, cautati in manualul imprimantei codul pentru latime dubla.)

Incercati si exemplul de mai jos:

```
10 OPEN #5;"b"
20 OPEN #6;"t"
30 PRINT #5; CHR$ 14;
40 LIST #6
50 CLOSE #5: CLOSE #6
```

Acest exemplu ar trebui sa produca un listing pe latime dubla.

INTRUCTIUNEA MOVE

Pina acum am facut transferuri de date de la un program catre un canal sau invers. Instructiunea MOVE va permite sa mutati date de la un canal la altul. De exemplu, pentru a muta date de la claviatura la ecran, introduceti:

```
10 MOVE #1 TO #2
```

apoi:

```
RUN
```

Orice veti introduce de la claviatura va apare pe ecran. Dar veti descoperi ca apasarea lui BREAK nu face decit sa tipareasca un spatiu pe ecran. Pentru a iesi din aceasta capcana, apasati CR pina ce ajungeti la ultima linie din ecran. Apoi, raspundeti cu BREAK la intrebarea scroll? (Ar trebui ca pe viitor sa evitati sa mutati date de la claviatura la orice alta cale pentru ca s-ar putea sa nu mai reusiti sa iesiti din instructiunea MOVE.)

Instructiunea MOVE se mai poate utiliza si pentru a examina fisierele memorate pe floppy. De exemplu, daca mai aveti pe floppy fisierul "Numere" (vezi pagina 7), ii puteti examina continutul cu instructiunea:

```
10 MOVE "d";1;"Numere" TO #2
```

(Remarcati ca nu trebuie sa deschideti/inchideti (OPEN/CLOSE) fisierul, MOVE face singur lucrul acesta.)

De asemenea, pentru a face o copie a programului "Numere" introduceti:

```
10 MOVE "d";1;"Numere" TO "d";1;"Numere 2"
```

In acest caz, MOVE deschide o cale pentru a citi din fisierul existent ("Numere") si o alta cale pentru a scrie in fisierul nou creat ("Numere 2"). Apoi citeste datele din "Numere" si le scrie in "Numere 2". Apoi inchide ambele cai.

MOVE va functiona atit cu numere de cale (ca de ex. #4), cit si cu specificatoare de canale (ca de ex. "d";1;"Numere"). Caile standard #1, #2 si #3 nu pot fi insa specificate cu numele consacrate K, S si P.

Puteti face o copie de siguranta a fisierului "Numere" pe alt disc folosind:

```
10 MOVE "d";1;"Numere" TO "d";2;"Numere 2"
```

Instructiunea MOVE poate fi folosita si pentru a trimite fisiere catre imprimanta. Daca aveti o imprimanta legata la interfata seriala, introduceti:

```
10 FORMAT "t",viteza
20 OPEN #4,"t"
30 MOVE "d";1;"Numere" TO #4
```

Programul Printer Server

Programul permite unui HC85 cuplat la o retea sa controleze o imprimanta seriala. Imprimanta poate fi folosita de toate calculatoarele cuplate la retea. Acest program este util daca, de exemplu, un grup de utilizatori de HC85 poseda o singura imprimanta seriala pe care vor sa o imparta. Se arata totodata o utilizare mai deosebita pentru instructiunea MOVE.

Calculatorul folosit ca Printer Server trebuie sa fie intotdeauna statia 64, si trebuie intotdeauna sa faca legatura cu statia 62 (care este o statie speciala de stabilire de contact). Astfel, statia emitatoare foloseste temporar statia 62, si trimite numarul sau real de statie, de la care va muta apoi un fisier spre canalul t. Pentru a stabili un program Printer Server introduceti:

```
10 FORMAT "n";64
20 OPEN #4;"n";62: INPUT #4;a$: CLOSE #4
30 MOVE "n";CODE a$ TO "t"
40 OPEN #4;"b": PRINT #4;CHR$ 12: CLOSE #4: RUN
```

(Linia 40 trimite un avans de pagina.)

Programul de mai jos este cel folosit de emitator. Mai intii, emitatorul stabileste temporar statia 62. Apoi se emite numarul real de statie al emitatorului. Apoi statia emitatorului revine la numarul sau real. In final, linia 60 trimite datele care trebuiesc imprimate (in acest caz listingul).

```
10 LET statia=numar
                 |
                 +------  introduceti aici numarul de
                          statie al HC-ului local

20 FORMAT "n";62
30 OPEN #4;"n";64: PRINT #4;CHR$ statia: CLOSE #4
40 FORMAT "n";statia
50 OPEN #4;"n";64
60 LIST #4
70 CLOSE #4
```

INSTRUCTIUNILE CLEAR # SI CLS

Se recomanda folosirea instructiunilor CLS # si CLEAR # in prima linie a oricarui program care foloseste interfata 1.

Instructiunea CLEAR

Asa cum instructiunea CLEAR sterge toate variabilele definite (operatie care se executa in mod automat si la RUN), instructiunea CLEAR # sterge toate canalele si caile definite prin program, efectuind urmatoarele operatii:

- decupleaza toate caile de la canalele deschise de catre utilizator
- elibereaza spatiul de memorie ocupat de aceste canale. (Zona CHANS va contine numai canalele predefinite "k", "s" si "p".)
- cupleaza caile #0, #1, #2 si #3 la canalele standard.
- trece toate discurile in starea R/W (vezi mai jos).

Nu trebuie sa se confunde efectul instructiunii CLEAR # cu efectul inchiderii prin CLOSE # a tuturor cailor. Spre deosebire de CLOSE #, instructiunea CLEAR # abandoneaza pur si simplu datele care se gasesc in canale. Daca, spre exemplu, se sterge prin CLEAR # un canal de disc prin care s-a scris intr-un fisier, datele din ultimul buffer vor fi pierdute, si mai grav, nici datele care au fost deja scrise pe disc nu vor fi accesibile pentru citire.

Mesajul de eroare "Disk 'R/O'" si CLEAR

Pentru a proteja datele inscrise pe floppy, interfata 1 utilizeaza o metoda de a preveni erorile datorate schimbarilor incorecte de floppy.

Daca interfata detecteaza o schimbare de suport intr-unul din minidrive-urile cu care a lucrat de la ultimul NEW sau CLEAR #, ea trece in mod automat discul respectiv in modul 'R/O' (numai citire).

Daca dupa o astfel de schimbare se incearca o operatie care necesita scrierea de date pe acel disc, se va obtine mesajul de eroare "Disk 'R/O'".

Pentru a corecta aceasta situatie, trebuie executata instructiunea CLEAR # imediat inainte de orice schimbare de suport intr unul din Minidrive-uri.

Instructiunea CLS

Efectele acestei instructiuni sint similare cu executia comenzilor:

```
PRINT;: BORDER 7: PAPER 7: INK 0: CLS
```